

Notice of Allowability

Application No.

10/789,012

Examiner

Phillip H. Nguyen

Applicant(s)

PULLARA, SAM

Art Unit

2191

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance (PTOL-85) or other appropriate communication will be mailed in due course. **THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS.** This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

1. ☒ This communication is responsive to amendment filed on 11/01/2007.
2. ☒ The allowed claim(s) is/are 2, 7, 12, 17, 22, 27, 32 and 34.
3. ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
 - a) ☐ All b) ☐ Some* c) ☐ None of the:
 1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____
 3. ☐ Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

* Certified copies not received: _____

Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application.
THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.

4. ☐ A SUBSTITUTE OATH OR DECLARATION must be submitted. Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL PATENT APPLICATION (PTO-152) which gives reason(s) why the oath or declaration is deficient.
 5. ☐ CORRECTED DRAWINGS (as "replacement sheets") must be submitted.
 - (a) ☐ including changes required by the Notice of Draftsperson's Patent Drawing Review (PTO-948) attached
 - 1) ☐ hereto or 2) ☐ to Paper No./Mail Date _____
 - (b) ☐ including changes required by the attached Examiner's Amendment / Comment or in the Office action of Paper No./Mail Date _____
- Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the front (not the back) of each sheet. Replacement sheet(s) should be labeled as such in the header according to 37 CFR 1.121(d).
6. ☐ DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Attachment(s)

1. ☐ Notice of References Cited (PTO-892)
2. ☐ Notice of Draftperson's Patent Drawing Review (PTO-948)
3. ☐ Information Disclosure Statements (PTO/SB/08),
Paper No./Mail Date _____
4. ☐ Examiner's Comment Regarding Requirement for Deposit
of Biological Material
5. ☐ Notice of Informal Patent Application
6. ☒ Interview Summary (PTO-413),
Paper No./Mail Date _____
7. ☒ Examiner's Amendment/Comment
8. ☒ Examiner's Statement of Reasons for Allowance
9. ☐ Other _____

DETAILED ACTION

1. This action is in response to the amendment filed on 11/01/2007.
2. Claims 2, 7, 12, 17, 22, 27, 32 and 34 are allowed.
3. Claims 1, 3-6, 8-11, 13-16, 18-21, 23-26, 28-31 and 33 have been canceled.

Examiner's Amendment

4. An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

Authorization for this examiner's amendment was given in a telephone interview with Mr. Hwang, Reg. No. 56,206 on 11/20/2007 to dependent claims 2, 7, 12, 17, 22, 27, 32 and 34, which are now independent claims. The examiner's amendment was necessitated to further clarify the claimed limitations of the independent claims.

The Applicant has been amended as follows:

2. (Currently Amended) A computer system comprising at least one processor to migrate a software application from one application server to a different application server, comprising:

a first application server that includes an application deployed thereon;

a preprocessor server component on the first application server used to interrogate the functionality of the deployed application, the application's deployment

information and any dependencies included therein, generate a new configuration information and communicate the new configuration information to a second application server that is used in deploying the application at the second application server;

wherein said preprocessor performs the step of:

interrogating the deployed application at the first application server to find all naming and directory interface entities present in the application;

determining which of said naming and directory interface entities will be realized at runtime; and,

parsing through both an application-side list, and a server-side list, and locating dependencies that correlate with one another; and

wherein the application-side list defines enterprise beans used in the application and resources dependent thereon, and the server-side list defines management interfaces used by the application, data source, and messaging service queues.

7. (Current amended) A method for migrating a software application from one software application server to a different software application server, comprising the step of:

providing a first application server that includes a preprocessor and an application deployed thereon;

using the preprocessor to interrogate the functionality of the application, the application's deployment information and any dependencies included therein, generate new configuration information and communicate the new configuration information to a

Art Unit: 2191

second application server that is used in deploying the application at the second application server;

wherein said preprocessor performs the step of:

interrogating the deployed application at the first application server to find all naming and directory interface entities present in the application;

determining which of said naming and directory interface entities will be realized at runtime; and,

parsing through both an application-side list, and a server-side list, and locating dependencies that correlate with one another; and

wherein the application-side list defines enterprise beans used in the application and resources dependent thereon, and the server-side list defines management interfaces used by the application, data source, and messaging service queues.

12. (Currently Amended) A computer readable medium including instructions stored thereon which when executed cause the computer to perform the step of:

providing a first application server that includes an application deployed thereon and a preprocessor;

interrogating the application's functionality, the application's deployment information and any dependencies included therein using the preprocessor, generating new configuration information, and communicating the new configuration information to a second application server that is used in deploying the application at the second application server;

wherein said preprocessor performs the steps of:

interrogating the deployed application at the first application server to find all
naming and directory interface entities present in the application;

determining which of said naming and directory interface entities will be realized
at runtime; and,

parsing through both an application-side list, and a server-side list, and locating
dependencies that correlate with one another;

communicating the new configuration information that is used in deploying the
application on said second application server; and

wherein the application-side list defines enterprise beans used in the application
and resources dependent thereon, and the server-side list defines management
interfaces used by the application, data source, and messaging service queues.

17. (Currently Amended) A computer system for readily deploying software applications
from a first server to a second server, comprising:

a first server having an application deployed thereon;

a second server adapted to receive said application;

a preprocessor on said first server that interrogates the application's functionality,
the application's deployment information as deployed on said first server, and any
dependencies included therein, and generates and communicates a new configuration
information, that is used in deploying the application at said second server;

wherein said preprocessor performs the steps of:

interrogating the deployed application at the first application server to find all naming and directory interface entities present in the application;

determining which of said naming and directory interface entities will be realized at runtime; and,

parsing through both an application-side list, and a server-side list, and locating dependencies that correlate with one another;

wherein the application-side list defines enterprise beans used in the application and resources dependent thereon, and the server-side list defines management interfaces used by the application, data source, and messaging service queues.

22. (Currently Amended) A method for readily deploying software applications from a first server to a second server, comprising the steps of:

providing a first server having an application deployed thereon and a preprocessor;

providing a second server adapted to receive said application;

interrogating the application's functionality, the application's deployment information as deployed on said first server, and any dependencies included therein using the preprocessor, and generating and communicating a new configuration information, that is used in deploying the application at said second server;

wherein said preprocessor performs the steps of:

interrogating the deployed application at the first application server to find all naming and directory interface entities present in the application;

determining which of said naming and directory interface entities will be realized at runtime; and,

parsing through both an application-side list, and a server-side list, and locating dependencies that correlate with one another;

communicating the new configuration information that is used in deploying the application on said second application server; and

wherein the application-side list defines enterprise beans used in the application and resources dependent thereon, and the server-side list defines management interfaces used by the application, data source, and messaging service queues.

27. (Currently Amended) A computer readable medium including instructions stored thereon which when executed cause the computer to perform the steps of:

providing a first server having an application deployed thereon and a preprocessor;

providing a second server adapted to receive said application;

interrogating the application's functionality, the application's deployment information as deployed on said first server, and any dependencies included therein using the preprocessor, and generating and communicating a new configuration information, that is used in deploying the application at said second server;

wherein said preprocessor performs the steps of:

interrogating the deployed application at the first application server to find all naming and directory interface entities present in the application;

determining which of said naming and directory interface entities will be realized at runtime; and,

parsing through both an application-side list, and a server-side list, and locating dependencies that correlate with one another;

communicating the new configuration information that is used in deploying the application on said second application server; and

wherein the application-side list defines enterprise beans used in the application and resources dependent thereon, and the server-side list defines management interfaces used by the application, data source, and messaging service queues.

32. (Currently Amended) The computer system of claim 2 wherein the deployment information is sifted to see which application entities will be realized at runtime.

34. (Currently Amended) The method of claim 7 wherein the deployment information is sifted to see which application entities will be realized at runtime.

Examiner's Statement of Reasons for Allowance

5. The following is an examiner's statement of reasons for allowance:

The prior art of record, i.e., Chen et al. (US 2003/0188036), taken alone or in combination with other prior art fails to teach or reasonably suggest *interrogating the deployed application at the first application server to find all naming and directory interface entities present in the application, determining which of said naming and*

directory interface entities will be realized at runtime, parsing through both an application-side list, and a server-side list, and locating dependencies that correlate with one another and wherein the application-side defines enterprise beans used in the application and resources dependent thereon, and the server-side list defines management interface used by the application, data sources, and messaging service queues as recited in independent claim 2. Similar concepts are found in independent claims 7, 12, 17, 22 and 27.

Any comments considered necessary by applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue fee. Such submissions should be clearly labeled "Comments on Statement of Reasons for Allowance."

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Phillip H. Nguyen whose telephone number is (571) 270-1070. The examiner can normally be reached on Monday - Thursday 10:00 AM - 3:00 PM EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Wei Y. Zhen can be reached on (571) 272-3708. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Art Unit: 2191

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

PN

11/21/2007



WEI ZHEN
SUPERVISORY PATENT EXAMINER